WEST Search History

| Hide Items | Restore | Clear | Cancel |

DATE: Thursday, July 08, 2004

Hide?	Set Name	Query	Hit Count
		DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ	
☐	L21	L20 NOT l5	25
☐	L20	L19 NOT l13	25
☐	L19	19991222	25
☐	L18	((publishig adj3 subsribing) or (publish adj3 subsribe) or PUB/SUB or PUBL/SUBS or publish/subscribe).ti.	56
☐	L17	(browser near9 (web adj3 server) near9 (application adj3 server) near9 request) and ((publishing adj3 subsribing) or (publish adj3 subsribe) or PUB/SUB or PUBL/SUBS or publish/subscribe)	3
☐	L16	(browser near9 (web adj3 server) near9 (application adj3 server) near9 request) same ((publishing adj3 subsribing) or (publish adj3 subsribe) or PUB/SUB or PUBL/SUBS or publish/subscribe)	0
☐	L15	browser near9 (web adj3 server) near9 (application adj3 server) near9 request	103
☐	L14	browser near9 (web adj3 server) near9 (application adj3 server)	517
☐	L13	L11 and (content or channel)	49
☐	L12	L11 and l1	37
☐	L11	L10 and (webcasting or webcast or broadcast or broadcasting)	50
☐	L10	19991222	127
☐	L9	((publishing adj3 subsribing) or (publish adj3 subsribe) or PUB/SUB or PUBL/SUBS or publish/subscribe)	452
☐	L8	((publishig adj3 subsribing) or (publish adj3 subsribe) or PUB/SUB or PUBL/SUBS or publish/subscribe)	452
☐	L7	(convert or conversion or transform or transforming) near8 (requst or message) near8 ((publishig adj3 subsribing) or (publish adj3 subsribe) or PUB/SUB or PUBL/SUBS or publish/subscribe)	0
☐	L6	L5 and l1	0
☐	L5	19991222	12
☐	L4	content-based adj2 (message or request)	33
☐	L3	19991222	2
☐	L2	subject-based adj2 (message or request)	5
☐	L1	information adj3 retrieval	21610

END OF SEARCH HISTORY

☐ Generate Collection

L5: Entry 12 of 12   File: USPT   May 2, 2000

DOCUMENT-IDENTIFIER: US 6058389 A
TITLE: Apparatus and method for message queuing in a database system

Application Filing Date (1):
19971031

Detailed Description Text (34):
An application program can specify message content using abstract data types (ADTs)
based upon the rich typing system of an existing database system. When the
invention is implemented with an object-relational database management system, such
as Oracle8, the application program may use relational data types as well as user-
defined types. As a result of messages having strongly typed content, powerful
features are enabled. For example, the queuing system can be used to provide
content-based message routing whereby an external agent or application program
examines content of a message and routes the message to another queue based upon
its content.

Search Forms

Search Results

Help   ☐ Generate Collection

User Searches

Preferences 2 of 12   File: USPT   Nov 25, 2003

Logout

DOCUMENT-IDENTIFIER: US 6654355 B1
TITLE: Bridge for CAN to TCP/IP connection

<u>Application Filing Date</u> (1):
19991214

<u>Brief Summary Text</u> (6):
As shown in FIG. 1, a CAN-type network 11a provides for communication of pre-
determined messages between stations 12a (nodes of the CAN network, each of which
are a control unit) interconnected in a linear bus structure by a CAN bus 14a. Each
CAN station is the peer of every other station. Instead of addressing a message to
another station by indicating the other station, a transmitting station indicates
to all other stations the content of the message using an identifier provided with
the message. In such <u>content-based addressing, each message</u> is broadcast to all
other, receiving stations, and each receiving station discards the message unless
the message is on a pre-determined acceptance list for the receiver.

Search Forms

Search Results

Help   ☐ ▓ Generate Collection ▓

·User Searches

Preferences L21 Entry 10 of 25   File: DWPI   Jan 1, 2002

Logout
DERWENT-ACC-NO: 2002-146801
DERWENT-WEEK: 200219
COPYRIGHT 2004 DERWENT INFORMATION LTD

TITLE: Content-based publish/subscribe system implementation method involves
multicasting events forwarded to interested consumers from specific brokers, by
different groups of brokers

☐ Generate Collection

L21: Entry 4 of 25   File: EPAB   Dec 6, 2000

PUB-NO: GB002350758A
DOCUMENT-IDENTIFIER: GB 2350758 A
TITLE: Message broker providing a publish/subscribe sevice and method of processing
messages in a publish/subscribe environment

PUBN-DATE: December 6, 2000

INVENTOR-INFORMATION:
NAME   COUNTRY
BIRD, COLIN LEONARD   GB
IBBOTSON, JOHN BRYAN   GB

INT-CL (IPC): H04 L 12/18; G06 F 9/46; G06 F 17/30; G06 F 17/60

ABSTRACT:

☐ Generate Collection

· L13: Entry 49 of 49   File: USPT   Sep 5, 2000

DOCUMENT-IDENTIFIER: US 6115693 A
TITLE: Quality center and method for a virtual sales and service center

Application Filing Date (1):
19980417

Brief Summary Text (45):
Another aspect of the present invention is that the monitor facilitates adjustments
to the Virtual Sales and Service Center for sending out broadcast/alert messages to
warn the physical locations of extremely high volume situations.

Detailed Description Text (15):
In many companies, the phone is a high volume access method. In the invention a
cloud 310 is established to source calls to the Virtual Sales and Service Center
300. All calls, including local numbers 312 and 1-800 numbers 314, are delivered to
this cloud 310. Within the cloud 310 are Voice Response Units (VRUs) 320 which play
a script that is heard by incoming customers placing calls to the Virtual Sales and
Service Center 300. The script played by the VRUs 320 enables a customer profile to
be identified. The content of the script is then personalized for each customer,
including matching the language being spoken by the caller. The VRUs 320 offer a
convenient navigation interface and can both meet customer requests directly or
initiate navigation to a resource that can handle the customer request. The VRUs
320 can also execute some cross-sell activities.

Detailed Description Text (33):
While intelligent routing provides rich functionality, the data it uses to make
decisions on call attributes must be processed very fast. Customer profiles,
customer accounts, and traditional account data will be accessed by a VRU 440 and
customer initiated VRU events will be passed to a service provider 410. The service
provider 410 maintains the business logic in channel independent applets.

Detailed Description Text (54):
A publish/subscribe manager 770 includes a statistics server 772 and a data
broadcast resource 774. Accordingly, this broadcast messaging capability enables
messages to be broadcast to the virtual environment based on the target recipient.
Recipients subscribe to messages intended for their user population.

Detailed Description Text (64):
send out broadcast/alert messages to inform agents of extremely high volume
situations.

Detailed Description Text (72):
In the event of system problems, or any circumstance where the Quality Center needs
to quickly contact the individual call centers, a broadcast messaging capability is
provided. The Quality Center reports on system and network performance 630 by
providing broadcast messaging capability. This allows the center to send out
messages to all call center personnel about the status of systems, anticipated
length of down time, status of closed branches, etc. The Quality Center may
broadcast messages based on target user populations via a publish/subscribe message

capability. Through this function, executives broadcast messages to the virtual environment based on the target recipient. Those recipients subscribe to messages intended for their user population.

CLAIMS:

42. The quality center of claim 1 wherein the messaging system provides immediate broadcast messages to resources in the plurality of physical locations.

43. The quality center of claim 42 wherein the broadcast messages are based upon target user populations via a publish/subscribe message.

45. A quality center for ensuring the satisfaction of operational efficiency of a Virtual Sales and Service Center and customer experience goals of customers accessing the Virtual Sales and Service Center, comprising:

a forecasting system for predicting contact volume for a plurality of physical locations forming a Virtual Sales and Service Center;

a monitor for monitoring contact traffic for the Virtual Sales and Service Center;

a controller for controlling network routing based upon the call volume predictions and the contact traffic monitoring; and

a processor for providing an interface between the forecasting system, the monitor and the controller and for servicing requests and response there between;

wherein the monitor facilitates adjustments to the Virtual Sales and Service Center for sending out broadcast/alert messages to warn the physical locations of extremely high volume situations.

Search Forms

Search Results

Help   □ ░░Generate Collection░░

User Searches

Preferences 48 of 49   File: USPT   Nov 28, 2000

Logout

DOCUMENT-IDENTIFIER: US 6154741 A
TITLE: Entitlement management and access control system

Application Filing Date (1):
19990408

Parent Case Text (2):
This application claims the benefit of U.S. Provisional Application No. 60/117,830,
filed on Jan. 29, 1999, and specifically incorporates the contents of that
application herein by reference.

Brief Summary Text (6):
As more business is done on computers, and particularly as more business is done
and information is exchanged across computer networks, access controls for
determining which computer users and software applications may obtain access to
which data or other computerized resources across these computer networks becomes
increasingly important. Access controls, for example, can control access to pages
on the World Wide Web, allowing differential content to be provided to different
groups of people, whether they are paying customers who pay for differing levels of
access, or to different groups of people who may have rights to differing levels of
confidential information. Access controls can also provide differing levels of
database access and transaction authorization as well as controlling the flow of
information that is broadcast or "pushed" over a computer network such as in
electronic publishing and message forwarding.

Brief Summary Text (19):
The entitlement manager system of the invention provides a new approach to access
control in complex systems and provides a dramatic advance over access control
lists by providing high speed resolution of dynamic access control rules. The
entitlement manager system thereby enables reliable charging of fees for content or
services in new ways without driving readers away by evaluating a user
characteristic at run time to determine entitlement to the content or services in
real time.

Detailed Description Text (56):
The term "client object," or more simply "client," refers to any object that uses
the resources of another object which is typically referred to as the "server
object" or "server." In one embodiment, the entitlement manager system of the
invention can be implemented as one or more server objects which can be accessed by
client objects seeking entitlement arbitration by the invocation of one or more
entitlement manager methods. In addition, objects and some other software
applications can communicate using a "publish/subscribe" protocol where an object
publishes information, sometimes called an "event," that is received by all other
objects that subscribe to that event. The entitlement manager system can control
the broadcast of events by arbitrating the entitlement of objects to the event--
that is, by determining which objects subscribe to the event.

☐ ▓▓ Generate Collection ▓▓

L13: Entry 47 of 49   File: USPT   Apr 10, 2001

DOCUMENT-IDENTIFIER: US 6216132 B1
TITLE: Method and system for matching consumers to events

Application Filing Date (1):
19971120

Brief Summary Text (7):
Another example of an event computing system is a distributed event system, also
known as a publish/subscribe system. A publish/subscribe system is a mechanism
where subscribers express interest in future information by some selection
criterion, publishers provide information, and the mechanism delivers the
information to all interested subscribers. Current publish/subscribe systems
organize information around groups (also called channels, subjects or streams).
Providers or publishers publish events to groups and consumers or subscribers
subscribe to all data from a particular group. Thus, in order to use a group based
publish/subscribe system, data must be pre-partitioned into groups. Although new
groups may be added to the system as it evolves, there is no mechanism to
reconfigure the existing groups in a system. Additionally, groups tend to partition
information along a single dimension. There is no elegant mechanism to support
applications that view data along another dimension.

Brief Summary Text (8):
One example of a publish/subscribe system is described in detail in U.S. Pat. No.
5,557,798, issued to Skeen et al. on Sep. 17, 1996, and entitled "Apparatus And
Method For Providing Decoupling Of Data Exchange Details For Providing High
Performance Communication Between Software Processes", which is hereby incorporated
herein by reference in its entirety. In U.S. Pat. No. 5,557,798, the publisher of
an event annotates each message with a group identifier called a subject and a
subscriber subscribes to a particular subject. Thus, if a subscriber is interested
in just a portion of the events having a given subject, it would have to receive
the entire subject and then discard the unwanted information.

Brief Summary Text (17):
In another aspect of the present invention, a method of publishing an event in a
publish/subscribe system is provided. The method includes, for example, providing,
by a publisher of the publish/subscribe system, an event to be published to one or
more subscribers of the publish/subscribe system. The event is independent of a
group association and lacks a group identifier. The event is published to the one
or more subscribers indicating interest in the event.

Brief Summary Text (18):
In a further aspect of the invention, a method of publishing an event in a
publish/subscribe system is provided. The method includes, for instance, indicating
interest in an event, by a first subscriber, using one attribute; indicating
interest in the event, by a second subscriber, using another attribute; and
publishing the event to the first and second subscribers.

Brief Summary Text (20):
In another aspect of the present invention, at least one program storage device

readable by a machine, tangibly embodying at least one program of instructions executable by the machine to perform a method of publishing an event in a publish/subscribe system is provided. The method includes, for instance, providing by a publisher of the publish/subscribe system an event to be published to one or more subscribers of the publish/subscribe system. The event is independent of a group association and lacks a group identifier. The event is published to the one or more subscribers indicating interest in the event.

Brief Summary Text (21):
In yet another aspect of the present invention, at least one program storage device readable by a machine, tangibly embodying at least one program of instructions executable by the machine to perform a method of publishing an event in a publish/subscribe system is provided. The method includes, for instance, indicating interest in an event, by a first subscriber, using one attribute; indicating interest in the event by a second subscriber, using another attribute; and publishing the event to the first and second subscribers.

Brief Summary Text (23):
In addition to the above, the present invention enables the construction of a publish/subscribe system based on content-based subscription. Content-based subscription allows consumers to specify a filter in its subscription rather than a group identifier. The filter enables the specification of one or more attributes for defining the search. Thus, the information sent to the subscriber is only that information requested by the subscriber.

Drawing Description Text (10):
FIG. 7 depicts one example of a publish/subscribe system incorporating and using the matching capability of the present invention; and

Detailed Description Text (5):
Consumer 120, which is, for instance, anyone or anything that is interested in data (e.g., an event), registers with registration interface 105 a filter and an action to take in response to an event. The filter is defined by a pattern, which describes those events of interest for a given consumer (e.g., a trigger in a database event system or a subscriber in a publish/subscribe system). In particular, the pattern includes one or more filter attributes (e.g., tests) that are used in matching the consumer to an event. Patterns define relations for each attribute, and therefore describe a (potentially infinite) set of events. The filter and its associated action is referred to herein as a pair.

Detailed Description Text (6):
Registration interface 105 assigns a unique identifier to the pair. In one example, the unique identifier is a numeric identifier provided by a counter of the registration interface. The registration interface delivers the filter, labeled with the identifier, to a filter interpreter 101 within matching engine 100. In one example, the delivery is made by passing parameters from registration interface 105 to filter interpreter 101. The registration interface also delivers the action, labeled with the identifier, to result unit 110. The delivery mechanism is similar to that described above. If the invention is being used to match triggers in a database, the action is the action when the filter matches an event; for a publish/subscribe system, the action is to deliver the matching event to the consumer.

Detailed Description Text (56):
Content-Based Subscription

Detailed Description Text (57):
In accordance with the principles of the present invention, a publish/subscribe system is constructed that does not rely on a prepartitioning of data into groups. Such a publish/subscribe system utilizes content-based subscription. Content-based

subscription is the ability of subscribers to specify interest in events based on operations limited only by the structure of the events and the operations supported by the pattern language.

Detailed Description Text (58):
Consider a stock market example where events for executed trades include the name of the security, the price, and the volume of shares being traded. Previously, to structure the stock market example using groups a group is assigned for every security and events are published to the group associated with the appropriate security. A user wishing to view this information along a different dimension is left without adequate support. For example, consider a trader who is interested in high volume transactions, e.g., those involving over 100,000 shares. In a group-based publish/subscribe system, such a subscriber is forced to subscribe to all groups and filter out itself all events except the few the subscriber needs.

Detailed Description Text (59):
With the content-based subscription of the present invention, one subscriber may specify interest in all events associated with a security, another may restrict itself to trades associated with a security, and a third subscriber may express interest in all trades over 100,000 shares regardless of security. While groups restrict subscriptions to a single dimension (i.e., the groups name), the technique of the present invention deals with multiple dimensions (in this example, security name, price, and number of shares).

Detailed Description Text (60):
FIG. 7 shows one example of a publish/subscribe system incorporating and using the content-based subscription of the present invention. Each provider 710 is coupled to a matching engine 700 on a provider node 750. Further, each consumer 720 (e.g., a subscriber) is coupled to registration interface 702. The matching engine and registration interface are, for example, similar to matching engine 100 and registration interface 105 in the system shown in FIG. 1. The publish/subscribe result unit is a special case of result unit 110 in the system shown in FIG. 1. The publish/subscribe result unit is provided with the identity of each consumer 720 registering a subscription. This identity includes enough information for the publish/subscribe result unit to communicate with the consumer through the distributed network. For each filter matched by matching engine 700 in response to an event from provider 710, the publish/subscribe result unit delivers the event to the subscribers registering the filter. Distribution of the event is through the distributed network and may be in a point-to-point manner (publish/subscribe result unit 701 communicates with each consumer 720 with a matching filter) or through a typical multicast, or broadcast mechanism. The distributed mechanism is dependent on the capabilities of the distributed network.

Other Reference Publication (15):
Bill Segall and David Arnold, "Elvin has left the building: A publish/subscribe notification service with quenching," Proceedings of AUUG97, Brisbane, Australia, Sep. 1997. Available from http://www.dstc.edu.au/Elvin/papers/AUUG97//AUUG97.html.

CLAIMS:

22. The method of claim 1, wherein said zero or more consumers are subscribers of a publish/subscribe system and said receiving comprises receiving, from a publisher of said publish/subscribe system, said event.

23. A method of publishing an event in a publish/subscribe system, said method comprising:

providing, by a publisher of said publish/subscribe system, an event to be published to one or more subscribers of said publish/subscribe system, said event being independent of a group association and lacking a group identifier;

determining said one or more subscribers interested in said event, said determining comprising using a search data structure to determine said one or more subscribers, said search data structure comprising a path having one or more levels, said one or more levels corresponding to one or more attributes, and wherein a value of at least one attribute is a don't care value meaning traversal of the path is guaranteed to proceed irrespective of whether another path is also followed, wherein when said another path is also followed said search data structure comprises a spatially parallel search structure; and

publishing said event to one or more subscribers indicating interest in said event.

26. The method of claim 24, wherein said one or more filters is located in said publish/subscribe system independent of said one or more subscribers.

52. At least one program storage device readable by a machine, tangibly embodying at least one program of instructions executable by the machine to perform a method of publishing an event in a publish/subscribe system, said method comprising:

providing, by a publisher of said publish/subscribe system, an event to be published to one or more subscribers of said publish/subscribe system, said event being independent of a group association and lacking a group identifier;

determining said one or more subscribers interested in said event, said determining comprising using a search data structure to determine said one or more subscribers, said search data structure comprising a path having one or more levels, said one or more levels corresponding to one or more attributes, and wherein a value of at least one attribute is a don't care value meaning traversal of the path is guaranteed to proceed irrespective of whether another path is also followed, wherein when said another path is also followed said search data structure comprises a spatially parallel search structure; and

publishing said event to one or more subscribers indicating interest in said event.

55. The at least one program storage device of claim 53, wherein said one or more filters is located in said publish/subscribe system independent of said one or more subscribers.

68. An article of manufacture, comprising:

at least one computer usable medium having computer readable program code means embodied therein for causing the publishing of an event in a publish/subscribe system, the computer readable program code means in said article of manufacture comprising:

computer readable program code means for causing a computer to provide an event to be published to one or more subscribers of said publish/subscribe system, said event being independent of a group association and lacking a group identifier;

computer readable program code means for causing a computer to determine said one or more subscribers interested in said event, said determining comprising using a search data structure to determine said one or more subscribers, said search data structure comprising a path having one or more levels, said one or more levels corresponding to one or more attributes, and wherein a value of at least one attribute is a don't care value meaning traversal of the path is guaranteed to proceed irrespective of whether another path is also followed, wherein when said another path is also followed said search data structure comprises a spatially parallel search structure; and

computer readable program code means for causing a computer to publish said event to one or more subscribers indicating interest in said event.

72. A system of publishing an event in a <u>publish/subscribe</u> system, said system comprising:

a publisher of said <u>publish/subscribe</u> system adapted to provide an event to be published to one or more subscribers of said <u>publish/subscribe</u> system, said event being independent of a group association and lacking a group identifier;

means for determining said one or more subscribers interested in said event, said means for determining comprising a search data structure useable in determining said one or more subscribers, said search data structure comprising a path having one or more levels, said one or more levels corresponding to one or more attributes, and wherein a value of at least one attribute is a don't care value meaning traversal of the path is guaranteed to proceed irrespective of whether another path is also followed, wherein when said another path is also followed said search data structure comprises a spatially parallel search structure; and

means for publishing said event to one or more subscribers indicating interest in said event.

<u>Previous Doc</u>   <u>Next Doc</u>   <u>Go to Doc#</u>

☐ ▓▓▓ Generate Collection ▓▓▓

L13: Entry 45 of 49   File: USPT   Oct 2, 2001

DOCUMENT-IDENTIFIER: US 6298455 B1
TITLE: Publish and subscribe data processing with failover using cascaded sequence
numbers

Abstract Text (1):
In a publish/subscribe data processing broker network having a plurality of broker
data processing apparatuses, each of which has an input for receiving published
messages directly from a publisher application and/or receiving subscription data
directly from a subscriber application, and each of which having a means for
receiving a published message on a first topic and assigning a broker-specific
sequence number to the received message; a first broker apparatus has: a software
unit for determining a failure of a neighboring broker apparatus which has provided
published messages on the first topic to the first broker apparatus; and a software
unit for sending historic resubscriptions with respect to the first topic to each
antecedent broker apparatus of the failed neighboring broker apparatus by using the
broker-specific sequence number corresponding to each antecedent broker apparatus.

Application Filing Date (1):
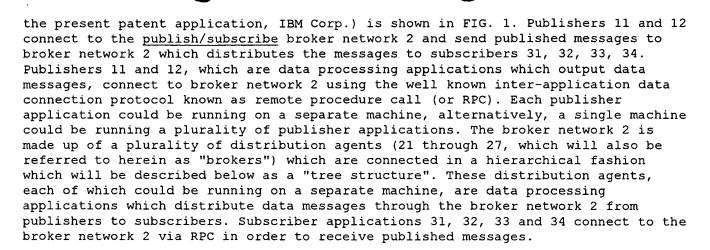19981229

Brief Summary Text (4):
Publish/subscribe data processing systems have become very popular in recent years
as a way of distributing data messages from publishing computers to subscribing
computers. The increasing popularity of the Internet, which has connected a wide
variety of computers all over the world, has helped to make such publish/subscribe
systems even more popular. Using the Internet, a World Wide Web browser application
(the term "application" or "process" refers to a software program, or portion
thereof, running on a computer) can be used in conjunction with the publisher or
subscriber in order to graphically display messages. Such systems are especially
useful where data supplied by a publisher is constantly changing and a large number
of subscribers needs to be quickly updated with the latest data. Perhaps the best
example of where this is useful is in the distribution of stock market data.

Brief Summary Text (5):
In such systems, publisher applications of data messages do not need to know the
identity or location of the subscriber applications which will receive the
messages. The publishers need only connect to a publish/subscribe distribution
agent process, which is included in a group of such processes making up a broker
network, and send messages to the distribution agent process, specifying the
subject of the message to the distribution agent process. The distribution agent
process then distributes the published messages to subscriber applications which
have previously indicated to the broker network that they would like to receive
data messages on particular subjects. Thus, the subscribers also do not need to
know the identity or location of the publishers. The subscribers need only connect
to a distribution agent process.

Brief Summary Text (6):
One such publish/subscribe broker network which is currently in use, and which has
been developed by the Transarc Corp. (a wholly owned subsidiary of the assignee of

the present patent application, IBM Corp.) is shown in FIG. 1. Publishers 11 and 12 connect to the publish/subscribe broker network 2 and send published messages to broker network 2 which distributes the messages to subscribers 31, 32, 33, 34. Publishers 11 and 12, which are data processing applications which output data messages, connect to broker network 2 using the well known inter-application data connection protocol known as remote procedure call (or RPC). Each publisher application could be running on a separate machine, alternatively, a single machine could be running a plurality of publisher applications. The broker network 2 is made up of a plurality of distribution agents (21 through 27, which will also be referred to herein as "brokers") which are connected in a hierarchical fashion which will be described below as a "tree structure". These distribution agents, each of which could be running on a separate machine, are data processing applications which distribute data messages through the broker network 2 from publishers to subscribers. Subscriber applications 31, 32, 33 and 34 connect to the broker network 2 via RPC in order to receive published messages.

Brief Summary Text (9):
The above-described publish/subscribe broker network architecture provides the advantage of central coordination of all published messages, since all publishers must connect to the same distribution agent (the root) in order to publish a message to the broker network hierarchy. For example, total ordering of published messages throughout the broker hierarchy is greatly facilitated, since the root can easily assign sequence numbers to each published message on a stream.

Brief Summary Text (11):
However, this FIG. 1 architecture also has the disadvantage of publisher inflexibility, since each publisher is constrained to publishing from the single root distribution agent, even when it would be much easier for a publisher to connect to a closer distribution agent. Accordingly, publish/subscribe software designers are beginning to consider architectures where publishers are allowed to publish messages directly to any distribution agent in the broker network. This clearly has the advantage of removing the above-mentioned constraint on publishers. However, as with any tradeoff, it presents other problems. For example, providing total ordering of published messages on a stream is very difficult, since there is no longer a central distribution agent (broker) from which all publishers must publish. In other words, the central coordination at the root that was present in the above-mentioned root-based architecture and that made failover so straightforward is no longer available.

Brief Summary Text (12):
Thus, there is a great need in the publish/subscribe data processing art for a way to carry out failover when the publish/subscribe broker network architecture does not provide for totally ordered streams.
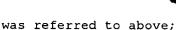
Brief Summary Text (14):
According to one aspect, the present invention provides In a publish/subscribe data processing broker network having a plurality of broker data processing apparatuses, each of which has an input for receiving published messages directly from a publisher application and/or receiving subscription data directly from a subscriber application, and each of which having a means for receiving a published message on a first topic and assigning a broker-specific sequence number to the received message; a first broker apparatus comprising:

Brief Summary Text (18):
Thus, with the present invention, by the use of the broker-specific sequence numbers, failover is made possible in a publish/subscribe broker network despite the fact that totally ordered streams are not present.

Drawing Description Text (3):
FIG. 1 shows the architecture of a prior art publish/subscribe broker network which

was referred to above;

Drawing Description Text (4):
FIG. 2 shows the architecture of a publish/subscribe broker network according to which the preferred embodiment of the present invention will be explained below;

Detailed Description Text (3):
Thus, whenever publisher 11 publishes a data message to stream "stock" the distribution tree structure of broker network 2 channels the message down through the root distribution agent 21, through any intermediary distribution agents (e.g., 22 in the example of FIG. 1) and through the leaf distribution agent 24 to the subscriber 32. This involves a series of RPC calls being made between each successive circle in the diagram of FIG. 1 connecting publisher 11 and subscriber 32 (i.e., 11 to 21, 21 to 22, 22 to 24 and 24 to 32).

Detailed Description Text (4):
FIG. 2 shows a different publish/subscribe architecture where publisher applications can publish messages to the broker network by directly communicating with any one of a plurality of distribution agents (also referred to herein as "brokers"). For example, publisher application 201 is shown communicating directly with Broker 12. There is no requirement in this architecture that all publisher applications communicate directly with a top (or root) distribution agent. Publisher application 201 can potentially communicate directly with any of the distribution agents shown in FIG. 2, in the described examples below it will be shown communicating directly with Broker 12.

Detailed Description Text (5):
Subscriber applications 202 and 203 would like to receive messages on the stream/topic that publisher application 201 is publishing on. Thus, subscriber applications 202 and 203 communicate directly with Brokers 1112 and 1221, respectively, to provide subscription data thereto informing the broker hierarchy of their desire to receive such published messages. Since the publisher application 201 is allowed to communicate directly with any of a plurality of distribution agents, the subscription data entered by the subscriber applications must be propagated throughout the broker network to each broker shown in FIG. 2. This way, no matter which distribution agent the publisher application 201 happens to communicate directly with, the published messages will be able to be routed to the subscriber applications 202 and 203. However, as mentioned above, the lack of central coordination of sequence numbers of published messages has prevented failover from being carried out with respect to this type of publish/subscribe broker network architecture.

Detailed Description Text (13):
In order to detect a broker failure, each broker transmits a heartbeat every so many (a predetermined number of) seconds. Failover begins if a larger interval of time has passed without a message being received from a given broker (e.g., broker B fails over to its antecedents (A1, A2, A3) if it has not heard from broker N in this time). The size of these intervals is a matter of tuning to avoid false failovers and should be broadcast to all brokers using an administrative function.

CLAIMS:

1. In a publish/subscribe data processing broker network having a plurality of broker data processing apparatuses,

each of which has an input for receiving published messages directly from a publisher application and/or receiving subscription data directly from a subscriber application, and

each of which having a means for receiving a published message on a first topic and

assigning a broker-specific sequence number to the received message;

a first broker apparatus comprising:

means for determining a failure of a neighboring broker apparatus which has provided published messages on the first topic to the first broker apparatus; and

means for sending historic resubscriptions with respect to the first topic to each antecedent broker apparatus of the failed neighboring broker apparatus by using the broker-specific sequence number corresponding to each antecedent broker apparatus.

6. In a <u>publish/subscribe</u> data processing broker network having a plurality of broker data processing apparatuses,

each of which has an input for receiving published messages directly from a publisher application and/or receiving subscription data directly from a subscriber application, and

each of which having a means for receiving a published message on a first topic and assigning a broker-specific sequence number to the received message;

a broker data processing method carried out at a broker data processing apparatus, the method comprising steps of:

determining a failure of a neighboring broker apparatus which has provided published messages on the first topic to the first broker apparatus; and

sending historic resubscriptions with respect to the first topic to each antecedent broker apparatus of the failed neighboring broker apparatus by using the broker-specific sequence number corresponding to each antecedent broker apparatus.

10. A computer program product stored on a computer readable storage medium for, when run on a computer, carrying out, in a <u>publish/subscribe</u> data processing broker network having a plurality of broker data processing apparatuses,

each of which has an input for receiving published messages directly from a publisher application and/or receiving subscription data directly from a subscriber application, and

each of which having a means for receiving a published message on a first topic and assigning a broker-specific sequence number to the received message;

a broker data processing method carried out at a broker data processing apparatus, the method comprising steps of:

determining a failure of a neighboring broker apparatus which has provided published messages on the first topic to the first broker apparatus; and

sending historic resubscriptions with respect to the first topic to each antecedent broker apparatus of the failed neighboring broker apparatus by using the broker-specific sequence number corresponding to each antecedent broker apparatus.

☐ Generate Collection

L13: Entry 16 of 49   File: USPT   Sep 9, 2003

DOCUMENT-IDENTIFIER: US 6618709 B1
TITLE: Computer assisted and/or implemented process and architecture for web-based
monitoring of energy related usage, and client accessibility therefor

Application Filing Date (1):
19981230

Brief Summary Text (46):
Optionally, the data recorder/translator unit includes a modem connectable to a
public switched telephone network or a wireless communications network, such as a
wireless telephone network. So configured, the data recorder/translator unit, in
operation, communicates with the server using standard communication methods such
as direct dial-up, Internet Protocol, or a publish/subscribe network communication
protocol.

Brief Summary Text (52):
Alternatively, the instant invention provides for a method of monitoring resource
usage via a global computer network. Resource usage data measured by a resource
meter is recorded, periodically or aperiodically, using a resource usage data
recorder/translator unit. The recorded resource usage data is published,
periodically or aperiodically, on a network, such as a local and/or global computer
network via a publish/subscribe network communication protocol using the data
recorder/translator unit. At least one local and/or global computer network server
subscribes to the published resource usage data, periodically or aperiodically. The
subscribed resource usage data is stored, periodically or aperiodically, to a
database. The recording step, the publishing step, the subscribing step, and the
storing step are repeated at regular or irregular time intervals.

Detailed Description Text (9):
Scalability: OPIS draws upon the technologies of the rich and diverse computing
environment, which ranges from individual workstations and departmental NT servers
all the way up to clusters of ultra high end servers, such as, Sun Enterprise
Servers, and mainframes, such as, IBM mainframes. OPIS optionally incorporates a
Hewlett-Packard Vantera-type information bus which utilizes the same messaging
technologies, i.e., publish/subscribe as Wall Street does to process several
hundred million trades a day. Other comparable alternatives may also be used.

Detailed Description Text (43):
Referring to FIG. 6, operation of the instant invention according to an alternative
embodiment is described as follows. In Step S200, a resource-metering data
recorder/translator unit, i.e., an intelligent meter 22, 24, 26, periodically or
aperiodically records resource usage. In Step S202, the data recorder/translator
unit publishes the resource usage data on a global computer network, for example,
the Internet 200, using standard publish/subscribe network communication protocol.
In Step S204, an Internet server 110 subscribes to the published resource usage
data. In Step S206, the resource usage data is stored in a searchable database 40.

Detailed Description Text (72):
Optionally, all data in the system is time and date stamped by customer and trade

channel. In other:words, all the data is optionally broken down into its least common denominator. This archival format allows for the re-construction and/or aggregation of any like type of information. Standard reporting may include line and bar graphs, HTML tables, comma delimited ASCII for import into any spreadsheet or summary bill report.

Detailed Description Text (74):
The system logs some and preferably all significant activity by a customer. The instant system includes standard tools for assessing demographics by content type to determine needs of different markets by tracking content usage of each market.

Detailed Description Text (97):
The present invention optionally and advantageously reads sub-meters. The sub-metered loads are defined as "Shredder1, 2, etc." along with their respective cost as shown in FIG. 6. Sub-meter reading is accomplished through the connection of pulse initiators from the sub-metered loads to the load data recorder. Those connections provide the necessary pulse information to the recorder and are identified on the recorder's channels by their end-use. The individual recorders are then accumulated to the enterprise level. At least substantially all data is stored at the channel level such that any configuration and/or amount of channels or submeters can be aggregated in any combination desired to produce aggregated load and cost profiles. Customers with existing regulatory agency-approved Direct Access meters would be "good to go" once the modem and phone line are installed.

Detailed Description Text (106):
Operations, maintenance and administration requirements may average, for example, two hours per week. There may be non-linear increasing returns to scale benefit for the maintenance and administration as the data acquisition system scales up in size. Flexibility is also present at this level. That is, it may be possible to proceed from initial access of a customer's load information to display of the customer's Internet solution, for example, in less than four hours. This margin may be reduced further by standard dynamic content creation development, whereby data may be available on the Internet as soon as the recorders are "plugged in."

Detailed Description Text (107):
The server capacity is, for example, approximately 500 customers, using a comparatively low-end workstation, and can be simply scaled via introduction of faster hardware, or alternatively across multiple machines. The crossover point from straight serial connection to publish/subscribe technology is in the 2,000 to 3,000 range.
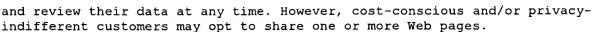
Detailed Description Text (114):
Utilizing HP Vantera-type technology, the instant invention may optionally communicate in at least the two phone and IP protocols, or by "publish/subscribe" technology. The latter technology permits the movement and dissemination of large amounts of information, that is, when simultaneous information transfer to or from, for example, in excess of 2,000 separate recorders. Vantera-type nodes and associated software provide for the `broadcast` and `subscription` of information to, from, and between the nodes over virtually any time interval.

Detailed Description Text (118):
The customer information is stored in the database by channel for each load recorder ID, so that the data is always available in its `least common denominator` format for any time period. This database optionally may be migrated to an ORACLE-type database as the economics warrant. To this extend, the SQL database is completely portable to an ORACLE-type database and has the capacity of, for example, more than 6,000 customers worth of information. Once the data is in the database, it is available or queryable by, for example, any practicable number of customers. Each customer optionally may have an individually secured page to access

and review their data at any time. However, cost-conscious and/or privacy-indifferent customers may opt to share one or more Web pages.

Detailed Description Text (122):
The HP Vantera-type communication services, for example, use an advanced commercially available messaging system for network efficient data transfer. Based on industry standard TCP/IP protocols, the messaging services may use publish/subscribe middleware to broadcast information to, from, and between servers and nodes in a true peer-to-peer fashion. Information transmission may be published from any source or pulled from any receiver in a manner similar to accessed information over the World Wide Web.

Detailed Description Text (149):
The system stores at least some of the data at the channel level and optionally creates aggregations as required by the customer. As long as individual channel data is available, it may be the data of choice. At least some of the information is aggregatable according to customer needs. Two or more channels of data may be brought together for aggregation as long as they are common in what they represent.

Detailed Description Text (170):
Optionally, a link to a database of tariff information may be provided. That information could be provided, for example, to the instant invention on a daily basis via FTP data transfer, for use in a billing module, for example, to feed a bill calculation engine, which would reside on the server. For this, a customers' unique recorder and channel identifier may be needed so that they can be integrated with the customer's load information and presented over the Internet to the customer.

CLAIMS:

13. The computer architecture according to claim 12, wherein said data recorder/translator unit includes a modem connectable to one of a public switched telephone network and a wireless communications network, and, in operation, communicates with said at least one server using one of direct dial-up, Internet Protocol, and a publish/subscribe network communication protocol.

20. A method of monitoring resource usage, via a global computer network, for at least one energy provider providing energy to a plurality of resource users, the method comprising the steps of: (a) recording, one of periodically and aperiodically, resource usage data measured by at least one resource meter, operatively connected to at least one remotely located resource consuming device, using a resource usage data recorder/translator unit; (b) publishing, one of periodically and aperiodically, the recorded resource usage data on a global computer network via a publish/subscribe network communication protocol, using the data recorder/translator unit; (c) subscribing, one of periodically and aperiodically, to the published resource usage data, using at least one global computer network server; (d) storing, one of periodically and aperiodically, the subscribed resource usage data to a database; (e) repeating said recording step (a), said publishing step (b), said subscribing step (c), and said storing step (d); (f) determining, simultaneous to said repeating step (e), whether a query from a user is received at the at least one server; (g) retrieving, when a query is received, resource usage data relevant to at least two resource users from the database; (h) aggregating the resource usage data for the at least two users; and (i) transmitting resource management information based on the aggregated resource usage data to the user one of at a global computer network site and in a downloadable data file for analysis by the user of the aggregated resource usage data collected by the at least two resource users provided by the at least one energy provider, the global computer network site and the downloadable data file being responsively connected to the global computer network server; and (j)

transmitting billing estimates to the user based on existing billing structures.

☐ ▓▓▓ Generate Collection ▓▓▓

L17: Entry 1 of 3   File: USPT   Jun 4, 2002

DOCUMENT-IDENTIFIER: US 6401085 B1
TITLE: Mobile communication and computing system and method

Detailed Description Text (171):
FIG. 18 discloses the detailed interaction between a consumer and the integrator
involving one supplier. The user accesses a Web Browser 1810 and requests product
and pricing information from the integrator. The request is sent from the user's
browser to the integrator's Web/Application Server 1820. The user's preferences and
personal information is obtained from an integrator's customer profile database
1830 and returned to the Web/Application server. The requested product information
is extracted from the supplier's product database 1840 and customized for the
particular customer. The Web/Application server updates the supplier's customer
information database 1850 with the inquiry information about the customer. The
product and pricing information is then formatted into a Web Page 1860 and returned
to the customer's Web Browser.

Detailed Description Text (275):
The Factual Layer is not aware of the Visual layer. This allows the visual metaphor
to change, without disrupting the underlying business domain model. The Behavioral
level mediates between the Factual and visual layers and should avoid very complex
interactions with either layer. Where possible, anonymous communications via a
Publish/Subscribe pattern is used to avoid further interdependencies between the
layers.

☐ ▓▓▓ Generate Collection ▓▓▓

L13: Entry 1 of 49   File: PGPB   Mar 27, 2003

DOCUMENT-IDENTIFIER: US 20030058277 Al
TITLE: A VIEW CONFIGURER IN A PRESENTATION SERVICES PATTERNS ENVIROMENT

Application Filing Date:
19990831

Detail Description Paragraph:
[0257] Sun's Java language has emerged as an industry-recognized language for
"programming the Internet." Sun defines Java as: "a simple, object-oriented,
distributed, interpreted, robust, secure, architecture-neutral, portable, high-
performance, multithreaded, dynamic, buzzword-compliant, general-purpose
programming language. Java supports programming for the Internet in the form of
platform-independent Java applets." Java applets are small, specialized
applications that comply with Sun's Java Application Programming Interface (API)
allowing developers to add "interactive content" to Web documents (e.g., simple
animations, page adornments, basic games, etc.). Applets execute within a Java-
compatible browser (e.g., Netscape Navigator) by copying code from the server to
client. From a language standpoint, Java's core feature set is based on C++. Sun's
Java literature states that Java is basically, "C++ with extensions from Objective
C for more dynamic method resolution."

Detail Description Paragraph:
[0258] Another technology that provides similar function to JAVA is provided by
Microsoft and ActiveX Technologies, to give developers and Web designers
wherewithal to build dynamic content for the Internet and personal computers.
ActiveX includes tools for developing animation, 3-D virtual reality, video and
other multimedia content. The tools use Internet standards, work on multiple
platforms, and are being supported by over 100 companies. The group's building
blocks are called ActiveX Controls, small, fast components that enable developers
to embed parts of software in hypertext markup language (HTML) pages. ActiveX
Controls work with a variety of programming languages including Microsoft Visual
C++, Borland Delphi, Microsoft Visual Basic programming system and, in the future,
Microsoft's development tool for Java, code named "Jakarta." ActiveX Technologies
also includes ActiveX Server Framework, allowing developers to create server
applications. One of ordinary skill in the art readily recognizes that ActiveX
could be substituted for JAVA without undue experimentation to practice the
invention.

Detail Description Paragraph:
[0345] To ensure that you are asking the right questions about the technology
architecture, you must refer to the Architecture Checklist (available from the
Content Finder). Questions will include:

Detail Description Paragraph:
[0445] Note: The Delivery Vehicle Framework presents a way to organize technology
architecture information. When presenting this type of content client, one may need
to tailor the information they present based on the client's background and the
terminology they are familiar with.

Detail Description Paragraph:
[0477] This is probably the main reason for selecting a Netcentric architecture.
Through appropriate use of a Netcentric architecture it is often possible to gain
exposure to new customers and markets. The client can often achieve significant
competitive advantage by providing new services and products to its customers. Also
this new channel makes it technically possible to develop a new generation of
"market-of-one" products, where each customer can repeatedly and easy customize a
product according to own preferences.

Detail Description Paragraph:
[0562] Definitions of Netcentric may vary. One is about reach and content.

Detail Description Paragraph:
[0569] Netcentric Computing also called Netcentric Architecture, Netcentric
Technology, etc. is an emerging architecture style which expands the reach of
computing both within and outside the enterprise. Netcentric enables sharing of
data and content between individuals and applications. These applications provide
capabilities to publish, interact or transact. Netcentric represents an evolution
of Client/Server which may utilize internet technologies to connect employees,
customers, and business partners.

Detail Description Paragraph:
[0645] Mapping Support--eliminate the need for applications to communicate directly
with the windowing system; rather, applications retrieve or display data by
automatically copying the contents of a window's fields to a copybook structure in
memory. These Services may also be used to automate the merging of application data
with pre-defined electronic form templates.

Detail Description Paragraph:
[0697] Web Browser Services allow users to view and interact with applications and
documents made up of varying data types, such as text, graphics, and audio. These
services also provide support for navigation within and across documents no matter
where they are located, through the use of links embedded into the document
content. Web Browser Services retain the link connection, i.e., document physical
location, and mask the complexities of that connection from the user. Web Browser
services can be further subdivided into: Browser Extension, Form, and User
Navigation.

Detail Description Paragraph:
[0708] Perhaps most significantly, HTML 4.0 increases authors' control over how
pages are organized by adding support for Cascading Style Sheets CSS Style sheets
contain directions for how and where layout elements such as margins, fonts,
headers, and links are displayed in Web pages. With CSS, authors can use
programming scripts and objects to apply multiple style sheets to Web pages to
create dynamic content. CSS can also be used to centralize control of layout
attributes for multiple pages within a Web site, thus avoiding the tedious process
of changing each page individually.

Detail Description Paragraph:
[0711] Unfortunately, the tremendous potential offered by DHTML is marred by
incompatible standards. At the heart of the DHTML debate is a specification called
the Document Object Model DOM. The DOM categorizes Web page elements--including
text, images, and links--as objects and specifies the attributes that are
associated with each object. The DOM makes Web document objects accessible to
scripting languages such as JavaScript and VisualBasic Script (VBScript), which can
be used to change the appearance, location, and even the content of those objects
in real-time.

Detail Description Paragraph:
[0716] A number of vendors plan to use XML as the underlying language for new Web

standards and applications. Microsoft uses XML for its <u>Channel</u> Definition Format, a Web-based "push" <u>content</u> delivery system included in Internet Explorer 4.0. Netscape will use XML in its Meta <u>Content</u> Framework to describe and store metadata, or collections of information, in forthcoming versions of Communicator. XML is currently playing an important role the realm of electronic commerce via the Open Financial Exchange, an application developed by Microsoft, Intuit, and CheckFree for conducting electronic financial transactions. Similarly, HL7, a healthcare information systems standards organization, is using XML to support electronic data interchange EDI of clinical, financial, and administrative information (http://www.mcis.duke.edu/standards/HL7/sigs/s- gml/index.html).

Detail Description Paragraph:
[0719] In order to create 3-D worlds and objects with VRML, users need a VRML editor such as Silicon Graphics' Cosmo Worlds (http://cosmo.sgi.com/products/studio/worlds). To view VRML <u>content,</u> users need either a VRML browser or a VRML plug-in for standard HTML browsers. Leading VRML plug-ins include Cosmo Player from Silicon Graphics (http://vrml.sgi.com/cosmoplayer), Liquid Reality from Microsoft's DimensionX subsidiary (http://www.microsoft.com/dimensionx), OZ Virtual from OZ Interactive (http://www.oz.com/ov/main_bot.html), and WorldView from Intervista (http://www.intervista.com/products/worldview/i- ndex.html), These plug-ins can typically be downloaded for free from the Web.

Detail Description Paragraph:
[0722] The Web has come a long way since the codification of HTML 1.0. It has moved from simple text-based documents that included headings, bulleted lists, and hyperlinks to dynamic pages that support rich graphic images and virtual reality. So what next for the Web? The answer resides in a Synchronized Multimedia Integration Language (SMIL), a new markup language being developed by the W3C. SMIL will allow Web authors to deliver television-like <u>content</u> over the Web using less bandwidth and a simple text editor, rather than intricate scripting.

Detail Description Paragraph:
[0723] SMIL is based on XML and does not represent a specific media format. Instead, SMIL defines the tags that link different media types together. The language enables Web authors to sort multimedia <u>content</u> into separate audio, video, text, and image files and streams which are sent to a user's browser. The SMIL tags then specify the "schedule" for displaying those components by determining whether they should be played together or sequentially. This enables elaborate multimedia presentations to be created out of smaller, less bandwidth-consuming components.

Detail Description Paragraph:
[0734] Web Browsers require new or at least revised development tools for working with new languages and standards such as HTML, ActiveX and Java. Many browser <u>content</u> development tools are available. The following are several representative products:

Detail Description Paragraph:
[0779] DimensionX Liquid Reality--VRML 2.0 platform written in Java, which provides both a viewer for viewing VRML <u>content</u> and a toolkit of Java classes for creating powerful 3-D applications. It supports more than 250 classes for 3-D <u>content</u> creation.

Detail Description Paragraph:
[0827] Many of the Netcentric applications are <u>broadcast</u>-type applications, designed to market products and/or publish policies and procedures. Furthermore, there is now a growth of Netcentric applications that are transaction-type applications used to process a customers sales order, maintenance request, etc. Typically these type of applications require integration with a database manager. Database Services include:

Detail Description Paragraph:
[0909] Access Services support document creation, maintenance and retrieval. These services allow users to capture knowledge or content through the creation of unstructured information, i.e. documents. Access Services allow users to effectively retrieve documents that were created by them and documents that were created by others. Documents can be comprised of many different data types, including text, charts, graphics, or even audio and video.

Detail Description Paragraph:
[0913] Locating documents and content within documents is a more complex problem and involves several alternative methods. The Windows file manager is a simplistic implementation of a hierarchical organization of files and collection of files. If the user model of where documents should be stored and found can be represented in this way, the use of structure and naming standards can be sufficient. However, a hierarchical document filing organization is not suitable for many types of document queries (e.g., retrieving all sales order documents for over $1,000).

Detail Description Paragraph:
[0916] Full-text Search--searches repository contents for exact words or phrases and returns documents that match the search criteria. In order to facilitate Full-text Search, full-text indexes are constructed by scanning documents once and recording in an index file which words occur in which documents. Leading document management systems have full-text services built-in, which can be integrated directly into applications.

Detail Description Paragraph:
[0917] Context Search--searches repository contents for exact words or phrases. Also, searches for related words or phrases by using synonyms and word taxonomies. For example, if the user searches for auto, the search engine should look for car, automobile, motor vehicle, etc.

Detail Description Paragraph:
[0918] Boolean Search--searches repository contents for words or phases that are joined together using boolean operators (e.g., AND, OR, NOT). Same type of indexes are used for Boolean Search as for Full-Text Search.

Detail Description Paragraph:
[0926] Storage Services manage the document physical storage. Most document management products store documents as objects that include two basic data types: attributes and content. Document attributes are key fields used to identify the document, such as author name, created date, etc. Document content refers to the actual unstructured information stored within the document. Generally, the documents are stored in a repository using one of the following methods:

Detail Description Paragraph:
[0927] Proprietary database--documents (attributes and contents) are stored in a proprietary database (one that the vendor has specifically developed for use with their product).

Detail Description Paragraph:
[0928] Industry standard database--documents (attributes and contents) are stored in an industry standard database such as Oracle or Sybase. Attributes are stored within traditional database data types (e.g., integer, character, etc.); contents are stored in the database's BLOB (Binary Large Objects) data type.

Detail Description Paragraph:
[0929] Industry standard database and file system--Documents' attributes are stored in an industry standard database, and documents' contents are usually stored in the file-system of the host operating system. Most document management products use

this document storage method, because today, this approach provides the most flexibility in terms of data distribution and also allows for greater scalability.

Detail Description Paragraph:
[0941] Communications middleware can provide additional communications services that may be required by the applications. Additional services include dynamic message routing, guaranteed delivery, broadcasting, queuing, and priority delivery. These common services are usually provided in the communications middleware rather than addressing them in each application separately. Different communications middleware products provide different services. Additionally, many middleware packages, such as Tuxedo, provide OLTP functionality.

Detail Description Paragraph:
[1057] Audio services--Audio services allow components to interface with audio streams such as the delivery of music or radio content over data networks.

Detail Description Paragraph:
[1059] Combined Audio/Video services--Video and audio content is often delivered simultaneously. This may be accomplished by transferring separate audio and video streams or by transferring a single interleaved stream. Examples include video conferencing and television (traditional or interactive).

Detail Description Paragraph:
[1061] Streams content (audio, video, or both) to end users

Detail Description Paragraph:
[1064] Manages communications protocols to ensure smooth delivery of content

Detail Description Paragraph:
[1065] Manages library of stored content and/or manages generation of live content

Detail Description Paragraph:
[1066] Audio/Video services draw upon lower-level services such as streaming and IP Multicast in order to efficiently deliver content across the network.

Detail Description Paragraph:
[1152] Multicasting and Broadcasting

Detail Description Paragraph:
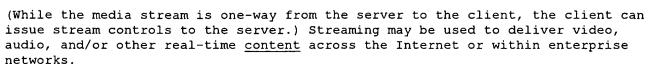[1243] Publish and subscribe (broadcasting)

Detail Description Paragraph:
[1253] TIB/Rendezvous' publish/subscribe technology is the foundation of TIBnet, TibCos solution for providing information delivery over intranets, extranets and the Internet. It is built upon The Information Bus.RTM. (TIB.RTM.) software, a highly scaleable messaging middleware technology based on an event-driven publish/subscribe model for information distribution. Developed and patented by TIBCO, the event-driven, publish/subscribe strategy allows content to be distributed on an event basis as it becomes available. Subscribers receive content according to topics of interest that are specified once by the subscriber, instead of repeated requests for updates. Using IP Multicast, TIBnet does not clog networks, but instead, provides for the most efficient real-time information delivery possible.

Detail Description Paragraph:
[1255] Streaming is the process of transferring time-sensitive data streams (e.g., video and/or audio) in real-time. Streaming differs from the other types of Core Messaging services in that it delivers a continuous, one-way stream of data, rather than the relatively short messages associated with RPC and Message-Oriented Middleware messaging or the large, batch transfers associated with File Transfer.

(While the media stream is one-way from the server to the client, the client can issue stream controls to the server.) Streaming may be used to deliver video, audio, and/or other real-time <u>content</u> across the Internet or within enterprise networks.

<u>Detail Description Paragraph</u>:
[1275] E-Mail takes on a greater significance in the modem organization. The E-Mail system, providing it has sufficient integrity and stability, can function as a key <u>channel</u> through which work objects move within, and between organizations in the form of messages and electronic forms. An E-Mail server stores and forwards E-Mail messages. Although some products like Lotus Notes use proprietary protocols, the following protocols used by E-Mail Services are based on open standards:

<u>Detail Description Paragraph</u>:
[1280] IMAP4--Internet Message Access Protocol, Version 4 (IMAP4) allows a client to access and manipulate electronic mail messages on a server. IMAP4 permits manipulation of remote message folders, called "mailboxes", in a way that is functionally equivalent to local mailboxes. IMAP4 also provides the capability for an off-line client to re-synchronize with the server. IMAP4 includes standards for message handling features that allow users to download message header information and then decide which e-mail message <u>contents</u> to download.

<u>Detail Description Paragraph</u>:
[1450] Filters--World Wide Web filters can prevent users from accessing specified <u>content</u> or Internet addresses. Products can limit access based on keywords, network addresses, time-of-day, user categories, etc.

<u>Detail Description Paragraph</u>:
[1459] Netscape Proxy Server--high-performance server software for replicating and filtering access to Web <u>content</u> on the Internet or an intranet. Provides access control, URL filtering, and virus scanning.

<u>Detail Description Paragraph</u>:
[1549] Distance Vector Protocols--each router periodically informs neighboring routers as to the <u>contents</u> of routing table (destination addresses and routing metrics); routing decisions based on the total distance and other "costs" for each path.

<u>Detail Description Paragraph</u>:
[1553] Link-State Protocols--each router periodically <u>broadcasts</u> changes to the routers and directly attached networks that it can talk with.

<u>Detail Description Paragraph</u>:
[1652] IP Switching--IP Switching is an emerging technology that can increase network throughput for streams of data by combining IP routing software with ATM switching hardware. With IP Switching, an IP switch analyzes each stream of packets directed from a single source to a specific destination, and classifies it as short- or long-lived. Long-lived flows are assigned ATM Virtual <u>Channels</u> (VCs) that bypass the IP router and move through the switching fabric at the full ATM line speed. Short-lived flows continue to be routed through traditional store-and-forward transfer.

<u>Detail Description Paragraph</u>:
[1670] multiplexing--A method of sharing physical media among nodes by consolidating multiple, independent <u>channels</u> into a single circuit. The independent <u>channels</u> (assigned to nodes, applications, or voice calls) can be combined in the following ways:

<u>Detail Description Paragraph</u>:
[1671] time division multiplexing (TDM)--use of a circuit is divided into a series

of time slots, and each independent <u>channel</u> is assigned its own periodic slot.

Detail Description Paragraph:
[1672] frequency division multiplexing (FDM)--each independent <u>channel</u> is assigned its own frequency range, allowing all <u>channels</u> to be carried simultaneously.

Detail Description Paragraph:
[1827] Subscribe and <u>Broadcast</u> supported

Detail Description Paragraph:
[1959] In creating ActiveX from OLE 2.0, Microsoft enhanced the framework to address some of the special needs of Web style computing. Microsofts Web browser, Internet Explorer, is an ActiveX container. Therefore, any ActiveX control can be downloaded to, and plugged into the browser. This allows for executable components to be interleaved with HTML <u>content</u> and downloaded as needed by the Web browser.

Detail Description Paragraph:
[2002] Push/Pull Services allow for interest in a particular piece of information to be registered and then changes or new information to be communicated to the subscriber list. Traditional Internet users "surf" the Web by actively moving from one Web page to another, manually searching for <u>content</u> they want and "pulling" it back to the desktop via a graphical browser. But in the push model, on which subscription servers are based on, <u>content</u> providers can <u>broadcast</u> their information directly to individual users' desktops. The technology uses the Internet's strengths as a two-way conduit by allowing people to specify the type of <u>content</u> they want to receive. <u>Content</u> providers then seek to package the requested information for automatic distribution to the user's PC.

Detail Description Paragraph:
[2005] Castanet from Marimba--distributes and maintains software applications and <u>content</u> within an organization or across the Internet, ensuring subscribers always have the most up-to-date information automatically.

Detail Description Paragraph:
[2314] A framework is a template for the implementation of a particular function (similar to a shell program). It usually embodies a known pattern (or group of patterns) in a specific technical environment. Frameworks are available from a number of third-party vendors, and they are also developed on projects. Developers are typically expected to customize and extend frameworks to meet their specific requirements, but this involves a tradeoff. Customizing and extending a framework may optimize its use, but the resulting framework tends to be less abstract, and therefore less reusable in other contexts. Examples of frameworks include: a framework for displaying an object and its properties in Smalltalk, a Java-specific framework for persisting data, and a messaging and <u>publish/subscribe</u> framework for DCOM.

Detail Description Paragraph:
[2388] multiple access <u>channels</u>

Detail Description Paragraph:
[2411] Multiple Access <u>Channels</u>

Detail Description Paragraph:
[2412] Component architectures are inherently service-oriented. Components provide their services through interfaces which consist of operations. Because components are independent pieces of software they can be reused by any number of applications. Thus, component-based architectures are well suited to environments that need to provide multiple application "personalities" or access <u>channels</u>. New personalities can be provided by creating a new user interface layer that reuses the existing business components.

Detail Description Paragraph:
[2661] A workcell's architect or frameworks developer can also help application developers better understand the architecture and use it consistently. Furthermore, the workcell architect serves as a good channel to feed new requirements--based on the application developers experiences--back to the architecture team.

Detail Description Paragraph:
[2907] Adaptable. Giving you freedom to deliver an application to a variety of user types through a variety of delivery channels with minimal impact to the application itself.

Detail Description Paragraph:
[3115] The Abstraction Factory pattern has a much broader applicability than just batch systems. It represents a way to encapsulate diversity such that only those parts of the system that need to understand the difference between two objects have to deal with those differences. To use a typical batch example, a file is a file is a file. Only those components that require knowledge of the contents of a file should need to deal with those contents in other than a very generic way.

Detail Description Paragraph:
[3214] Channeled Pipes. Perhaps the most generally useful form of a pipe is based on the CORBA Event Channel object, which can connect any number of Push/Pull Suppliers to any number of Push/Pull Consumers.

Detail Description Paragraph:
[3230] FIG. 58 illustrates a flowchart for a method 5800 for controlling access to data of a business object via an attribute dictionary. A plurality of attribute values for a business object are stored in an attribute dictionary in operation 5802. A plurality of attribute names are provided in the attribute dictionary for the stored attribute values in operation 5804. Next, in operation 5806, it is verified that a current user is authorized to either set or get one of the attribute values upon a request which includes the attribute name that corresponds to the attribute value. The attribute value in the attribute dictionary is obtained or updated if the verification is successful and an indicator is broadcast upon the attribute value being updated in operations 5808 and 5810.

Detail Description Paragraph:
[3235] Initially, this is straightforward. However, after all of the attribute setters and getters have been coded, the need may arise for an event to be broadcast each time an attribute is updated. The code for a simple setter would need to change to become:

Detail Description Paragraph:
[3247] if successful, broadcast and log the change

Detail Description Paragraph:
[3253] broadcasting

Detail Description Paragraph:
[3356] FIG. 71 illustrates a flowchart for a method 7100 for delivering service via a globally addressable interface. A plurality of interfaces are provided in operation 7102 and access is allowed to a plurality of different sets of services from each of the interfaces in operation 7104. Each interface has a unique set of services associated therewith. Each of the interfaces is named in operation 7106 with a name indicative of the unique set of services associated therewith. The names of the interfaces are then broadcast to a plurality of systems requiring service in operation 7108.

Detail Description Paragraph:

[3904] Communication from the activity to the View Configurer will be conducted through underline broadcasting (as described in the Observer pattern). In this manner, the activity doesn't know about the existence of the View Configurer, it listens to activity broadcasts such as when the controller starts up. This configurer can use the Observable Factory to get a handle to the activity instance.

Detail Description Paragraph:
[3909] Because the View Configurer has pre-registered an interest in the startup of activities, it will receive a broadcast message. In this step, the View Configurer should receive a minimum of two parameters:

Detail Description Paragraph:
[3919] The Observer Pattern (Gang of Four Pattern) describes how to provide visibility to other entities via a one to many relationship. A singleton activity factory will create new activity instances, and broadcast the startup of the new activities to the View Configurer.

Detail Description Paragraph:
[3920] An interface for the creation of activities is used in conjunction with the Observer Pattern. In this way, the startup of new activity instances can be broadcast to the View Configurer. This is described in the Factory Pattern (Gang of Four Pattern).

Detail Description Paragraph:
[3982] In the design of a stateful server, the LUW Context pattern facilitates the server process constructing domain objects at the request of the clients and maintaining these objects within a given context. Domain objects are entered into a registry with their appropriate context which the server maintains and updates when a request is received to create or delete an object. Each time a context is accessed then a notification is broadcast to the registry, regardless of a state change. With a simple context management, each time a context is referenced by a client a reference counter is incremented and similarly decrements when the reference is destroyed. Once the reference count returns to 0 then the context can be removed from the registry.

Detail Description Paragraph:
[4011] Let's take another scenario. Suppose you want to prevent any raised exception from bringing down your system, as least not without a fight. In some cases the error will be unrecoverable and there is not much you can do but release resources (locks, communication channels, . . .) and terminate. What caused the problem is going to be on the tops of the minds of the production support people, and yours when you get their call (always in the middle of the night). You could write the exception handling logic chunks for each exception type—remembering that each exception has its own interface and will require separate logic to handle each interface—for each exception, but now you have to handle all the exceptions in the system. Wouldn't it be nice to write one chunk of handling logic and be done with it?

Detail Description Paragraph:
[4553] The business object may also be sent to another context as at least one of a single focus of a window that is being created and a parameter in an explicit parameter-passing mechanism. Additionally, the copies of the business objects may be created from a same retrieved data stream. As a further option, receiving a request to make changes to a copy of the business object of one of the logical units of work and changing that copy of the business object may further include the broadcasting of the change to the other logical units of work.

Detail Description Paragraph:
[4557] However, an MVC-based OO architecture does not naturally support this requirement. With MVC, the domain model stores all data changes. Windows are merely

a view into this model, and they have little business data of their own. In addition, MVC model objects have no idea which views are using them. Instead, the model anonymously broadcasts its data changes, and all views on the model respond by updating themselves. This synchronizes windows with their business data. Thus, MVC allows multiple views to simultaneously display, and be refreshed by, a single copy of the model data.

Detail Description Paragraph:
[4605] However, when an LUW context successfully commits changes, it will have more current data than other contexts which it intersects. This up-to-date data can be broadcast and shared with the other contexts. These contexts can then decide to transparently incorporate the changes or not.

Detail Description Table CWU:
2 In theory . . . In practice . . . Flexible Making it possible to Making it possible to accom- quickly satisfy new busi- modate a new product line ness requirements by solely by updating the Product replacing or modifying component. certain components with minimal impact to others. Adaptable Making it easy to deliver an Making it easy to provide in- application to a variety of home access to customer user types through a variety account information by of delivery channels with developing only a new user minimal impact to the core interface while reusing application. existing components. Maintainable Making it easy to update an Making it easy to add a new application by reducing the customer attribute by isolating area of impact for most the change to one compon- changes. ent - the Customer compon- ent. Reusable Making it possible to Making it possible to as- quickly assemble unique semble an application at a and dynamic solutions from fraction of the cost because existing components. eight of the twelve compo- nents that are needed already exist. Integration Making it possible to reuse Making it possible to absorb Ready the functionality within newly acquired divisions by existing systems by wrap- "wrapping" their systems and ping them as components "plugging" them into the within new applications. enterprise infrastructure. Interoperable Making it possible to Making it possible to integrate request services across two applications built on platforms. different platforms. Scalable Making is easy to distribute Making it easy to accom- and reconfigure compo- modate the holiday crunch by nents to satisfy various running multiple copies of the transaction volumes. Order component across multiple servers.

Detail Description Table CWU:
6 public void setBalance(Float newBalance) { // keep track of my original balance, // for post-change processing, then do // some pre-processing to check // that the user has access rights Float oldBalance = myBalance; this.assertCanSetAttribute("Balance"); // finally update the balance, then // broadcast, set the Dirty Flag, // and log myBalance = newBalance; this.notifyChanged("Balance"); this.makeDirty( ); this.logChanged("Balance", oldBalance); }

Detail Description Table CWU:
12 . . . DATA DIVISION. FD FILE-STREAM-IN RECORD CONTAINS 20 CHARACTERS . . . WORKING-STORAGE SECTION. *** THIS COPYBOOK CONTAINS THE COMMON FORMAT OF THE *** CUSTOMER IN THE DATA STRUCTURE AND DATA TYPES 01 WS-COMMON-FORMAT-CUSTOMER 03 WS-COMMON-FORMAT-NAME PIC X(10). 03 WS-COMMON-FORMAT-SEX PIC X(7). 03 WS-COMMON-FORMAT-AGE PIC 999. *** THIS COPYBOOK IS THIS SYSTEMS VIEW OF A CUSTOMER 01 WS-CUSTOMER 03 WS-NAME PIC X(10). 03 WS-AGE PIC 999. 03 WS-SEX PIC X(10). . . . PROCEDURE DIVISION. . . . *** OPEN THE FILE STREAM AND PUT THE CONTENTS IN THE *** WS-COMMON-FORMAT-CUSTOMER COPYBOOK. OPEN FILE-STREAM-IN READ FILE-STREAM-IN INTO WS-COMMON-FORMAT- CUSTOMER AT-END CLOSE FILE-STREAM-IN END-READ. *** MOVE THE VALUES INTO FROM THE COMMON FORMAT INTO *** THE WS-CUSTOMER VARIABLES. MOVE WS-COMMON-FORMAT-SEX TO WS-SEX. MOVE WS-COMMON-FORMAT-AGE TO WS-AGE. MOVE WS-COMMON-FORMAT-NAME TO WS-NAME. . . . *** CALL A SQL MODULE TO SAVE THIS INFORMATION IN THE *** RELATIONAL DATABASE CALL "SAVE-CUSTOMER-IN-DATABASE" USING WS- CUSTOMER. . . .

STOP-RUN.

Detail Description Table CWU:
24 FD FILE-STREAM-IN RECORD CONTAINS 20 CHARACTERS . . . WORKING-STORAGE SECTION.
*** THIS COPYBOOK CONTAINS THE SHARED FORMAT OF THE *** CUSTOMER IN THE DATA
STRUCTURE AND DATA TYPES 01 WS-SHARED-FORMAT-CUSTOMER 03 WS-SHARED-FORMAT-NAME PIC
X(10). 03 WS-SHARED-FORMAT-SEX PIC X(7). 03 WS-SHARED-FORMAT-AGE PIC 999. *** THIS
COPYBOOK IS THIS SYSTEMS VIEW OF A CUSTOMER 01 WS-CUSTOMER 03 WS-NAME PIC X(10). 03
WS-AGE PIC 999. 03 WS-SEX PIC X(10). . . . PROCEDURE DIVISION. . . . *** OPEN THE
FILE STREAM AND PUT THE CONTENTS IN THE *** WS-SHARED-FORMAT-CUSTOMER COPYBOOK.
OPEN FILE-STREAM-IN READ FILE-STREAM-IN INTO WS-SHARED-FORMAT- CUSTOMER AT-END
CLOSE FILE-STREAM-IN END-READ. *** MOVE THE VALUES INTO FROM THE SHARED FORMAT INTO
*** THE WS-CUSTOMER VARIABLES. MOVE WS-SHARED-FORMAT-SEX TO WS-SEX. MOVE WS-SHARED-
FORMAT-AGE TO WS-AGE. MOVE WS-SHARED-FORMAT-NAME TO WS-NAME. . . . *** CALL A SQL
MODULE TO SAVE THIS INFORMATION IN THE *** RELATIONAL DATABASE CALL "SAVE-CUSTOMER-
IN-DATABASE" USING WS- CUSTOMER. . . . STOP-RUN.